NT30-DPS 网关应用说明书

一、NT30-DPS 网关概述

NT30-DPS 网关将串口协议转换为 Profibus-DP 从站，应用于将具备串口通讯协议的设备（如 PLC、仪器仪表、传感器等）接入 Profibus-DP 网络从而接受 Profibus-DP 主站的控制。NT30-DPS 包括四个订货号：

1、 NT30-DPS-RS2：RS232 至 Profibus-DP 从站转换

2、 NT30-DPS-RSI2：RS232 至 Profibus-DP 从站转换，RS232 端口电气隔离

3、 NT30-DPS-RS4：RS422/RS485 至 Profibus-DP 从站转换

4、 NT30-DPS-RSI4：RS422/RS485 至 Profibus-DP 从站转换，RS422/RS485 端口电气隔离

二、应用架构

NT30-DPS 在 Profibus-DP 总线上作为 Profibus-DP 从站运行，具有唯一的从站地址（可以在网关上设定），通过 Profibus-DP 的输入输出映像区接受 Profibus-DP 主站（如西门子的 CPU315-2DP）控制。NT30-DPS 网关通过串行总线可以连接多个串口协议设备（如 Modbus RTU 设备），应用架构如下图所示：

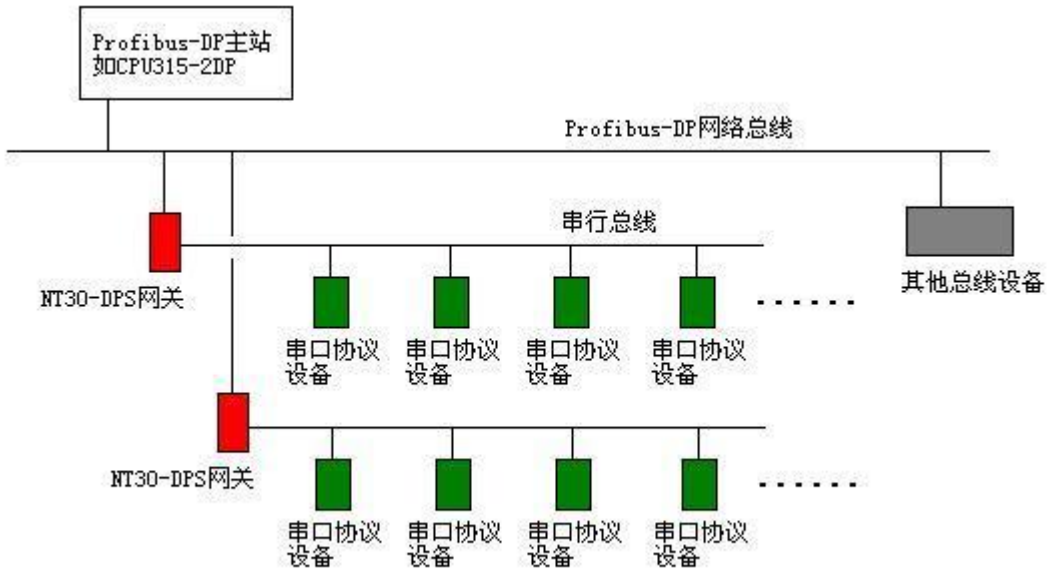


图1 NT30-DPS网关应用架构

注：一个 NT30-DPS 网关可以连接的串口协议设备的数量和所使用的协议固件相关，并且也受到 Profibus-DP 从站映像区大小的限制（NT30-DPS 的输入输出映像区最大为 244 个字节）。设计时如果读写的数据量较大可以选择多个网关解决。

三、协议固件

NT30-DPS 有三个协议固件可供选择，这些协议固件保存在随机光盘的 Firmware 文件夹中；网关在出厂时内部无预装固件，因此在使用网关前必须先下载相应的固件（通过 SYCON.net 软件，软件在随机光盘中）。三个协议固件如下：

1、NTDPSMBR.N34 ：ModbusRTU（主/从站）至 Profibus-DP 从站协议转换固件文件名

2、NTDPSASC.N34 ：自定义 ASCII 协议至 Profibus-DP 从站协议转换固件文件名

3、NTDPSNVR.N34 ：西门子 3964R 协议至 Profibus-DP 从站协议转换固件文件名

网关在不同协议固件下的工作模式区别：

1、 在 ModbusRTU 和 3964R 协议至 Profibus-DP 从站协议转换固件下，网关工作于 Memory-Memory 模式，即共享内存模式。在该模式下，网关根据用户配置的命令表（COMMAND 表，在 SYCON.net 软件中配置）自动循环执行每个命令，并且和 Profibus-DP 输入输出映像区直接交换数据；进一步说，对于读命令，网关会把读取的设备数据自动放入 Profibus-DP 的输入映像区供 Profibus-DP 主站读取；对于写命令，网关自动取出 Profibus-DP 的输出映像区数据并写入设备，Profibus-DP

主站只需要将待写数据放入 Profibus-DP 的输出映像区即可。Memory-Memory 模式如下图所示（实际输入输出映像区大小可以自定义）：

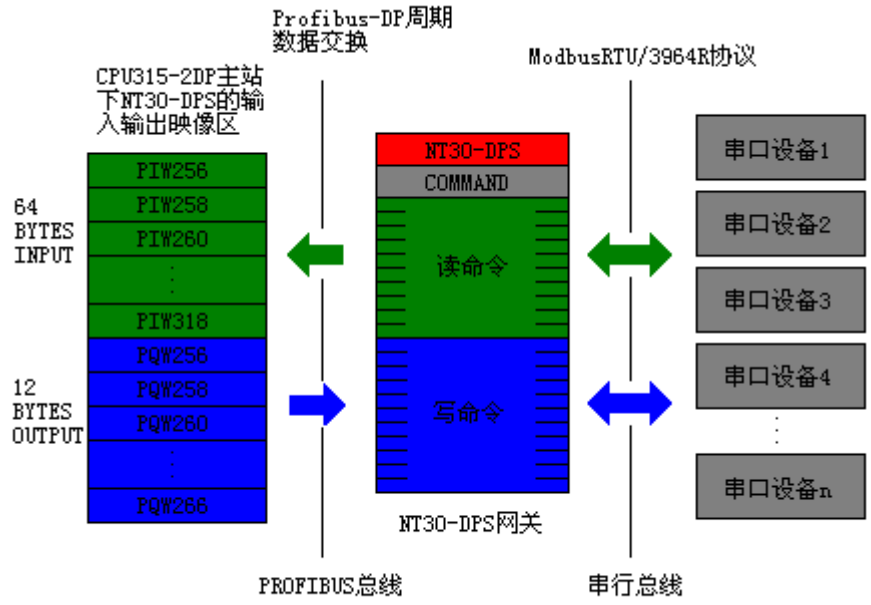


图2 Memory-Memory工作模式

2、 在自定义 ASCII 协议至 Profibus-DP 从站协议转换固件下，网关工作于 Transparent-Conversion 模式，即透明转换模式。在该模式下由于网关不清楚自定义协议的命令/数据帧格式，因此配置表中没有 COMMAND 命令表。在该模式下网关不会主动发起对任何端口（Profibus-DP 端口和串行总线端口）的通讯，网关必须在接收到任一端口方向来的数据帧后才开始进行协议转换。在透明转换模式下，来自串口设备的数据帧将无变化的进入 Profibus-DP 输入映像区，Profibus-DP 输出映像区的数据也将无变化的发送到串行总线上。那么，当网关接收到数据该如何通知 CPU 以及 CPU 如何要求网关将数据发送出去呢？NT30-DPS 网关通过输入输出映像区中的前两个字节（控制字）来解决。如下图所示：

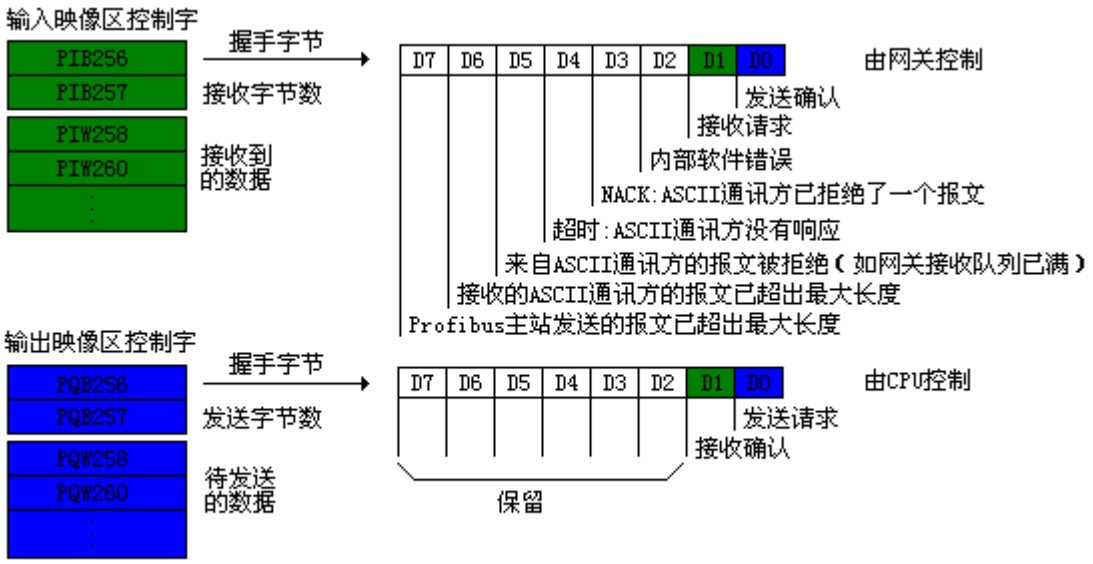


图3 ASCII固件下的握手信息

从上图可以看出，在 ASCII 固件下的发送/接收流程为：

1) CPU 发送数据：

CPU 将需要发送的协议数据放入输出映像区数据区，然后将整个协议数据帧长度放入 PQB257，最后取反[发送请求]，当网关监视到[发送请求]和[发送确认]的位值不相等时就将开始发送数据；当网关将数据全部发送到串行总线上后自动将[发

送确认]取反，以通知 CPU 数据已经发送完成；

2) CPU 接收数据：

当来自串口设备的数据被网关接收到后，网关自动将整个协议数据帧放入输入映像区的数据区，然后取反[接收请求]，CPU 可以监视[接收请求]的位值，当位值发生变化后就可以从输入映像区取出数据，然后取反[接收确认]以通知网关数据已经被取走（网关可以继续接收下一个数据帧）。

注意：[发送请求]和[发送确认]对应，[接收请求]和[接收确认]对应。

在实际应用中，我们可以通过在 CPU 中运行下面的指令来使得网关连续不断的发送/接收数据（在西门子 STEP7 的 OB1 中）：

```
A [接收请求]     //如果有接收请求则马上确认它
= [接收确认]
AN [发送确认]    //通过使[发送请求]和[发送确认]永远不相等来触发网关不断发送数据
= [发送请求]
```

四、网关接线

NT30-DPS 在应用时需要完成两个端口方向的接线：

1、 Profibus-DP 端口接线：

网关的 Profibus-DP 端口是一个 SubD 9 针母接口，它遵循 Profibus 接口标准，在连接时采用西门子的 Profibus 电缆和网络插头将网关连接到 Profibus-DP 总线上，同时在网关的地址拨码盘上设定网关的 Profibus-DP 从站地址。更改了从站地址后需要将网关重新上电。如果网关是 Profibus-DP 总线上的最后一个站点，请将网络插头的终端电阻设置为 ON。

2、 串行总线端口的接线：

串行总线端口在连接时请参考随机光盘中的接线说明书，请仔细核对串口设备的通讯端口接线图和网关 RS232/RS422/RS485 端口的接线图。

五、Modbus 固件下的配置和应用举例

将 ModbusRTU 设备接入 Profibus-DP 总线是 NT30-DPS 的典型应用，在这里我们通过 Modbus 仿真软件 ModSim32 来演示网关的应用，演示接线图如下：

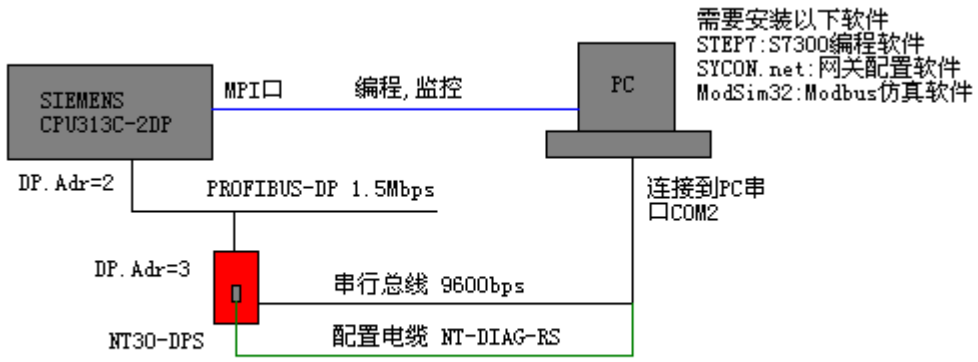


图4 NT30-DPS的Modbus应用演示

注：在这个例子中由于采用了计算机串口进行 Modbus 仿真，因此需要选用 NT30-DPS-RS2 网关，为描述简便，以下简称 NT30-DPS。

应用目的

通过运行 ModSim32 仿真软件仿真出三个 ModbusRTU 从站(从站地址为 2,3,4，网关的 ModbusRTU 地址为 1)，NT30-DPS 网关需要实现的通讯任务：

① 读取每个站点的线圈状态 00001-00016，共 16 个位，Modbus 功能号 FC1。

② 读取每个站点的输入状态 10001-10016，共 16 个位，Modbus 功能号 FC2。

③ 读取每个站点的保持寄存器 40001-40015，共 15 个寄存器，Modbus 功能号 FC3。

④ 读取每个站点的输入寄存器 30001-30015，共 15 个寄存器，Modbus 功能号 FC4。

⑤ 写入每个站点的线圈状态 00001-00016，共 16 个位，Modbus 功能号 FC5。

⑥ 写入每个站点的保持寄存器 40001，共 1 个寄存器，Modbus 功能号 FC6。

根据以上要求，我们计算一下网关的 Profibus-DP 输入输出映像区该设计成多大：

读取数据的总空间为 FC1，FC2，FC3，FC4 的读取空间和，FC1 读取 1 个字，FC2 读取 1 个字，FC3 读取 15 个字，FC4 读取 15 个字，共 32 个字，即每个站点需要读取 64 个字节，三个站点一共需要 64*3=192 个字节空间。也就是说 NT30-DPS 的 Profibus-DP 输入映像区应该是 192 个字节大小。

写数据的空间为 FC5 和 FC6 的写入空间和，FC5 写入 1 个字，FC6 写入 1 个字，共 2 个字，即每个站点需要写入 4 个字节，三个站点一共需要 4*3=12 个字节空间。也就是说 NT30-DPS 的 Profibus-DP 输出映像区应该是 12 个字节大小。

总结：NT30-DPS 的输入映像区为 192 字节，输出映像区为 12 字节。

以上任务 1 和任务 5 地址对应，任务 3 和任务 6 地址对应，这是为了能够验证读写是否正确：通过任务 5 写入的位值可以通过任务 1 读出来；通过任务 6 写入的 40001 寄存器值可以通过任务 3 读出来。

地址映射表

确定好网关的 Profibus-DP 输入输出映像区后所有 Modbus 变量地址也就确定了，我们设定输入输出映像区地址都从 256 开始，则如下图所示：

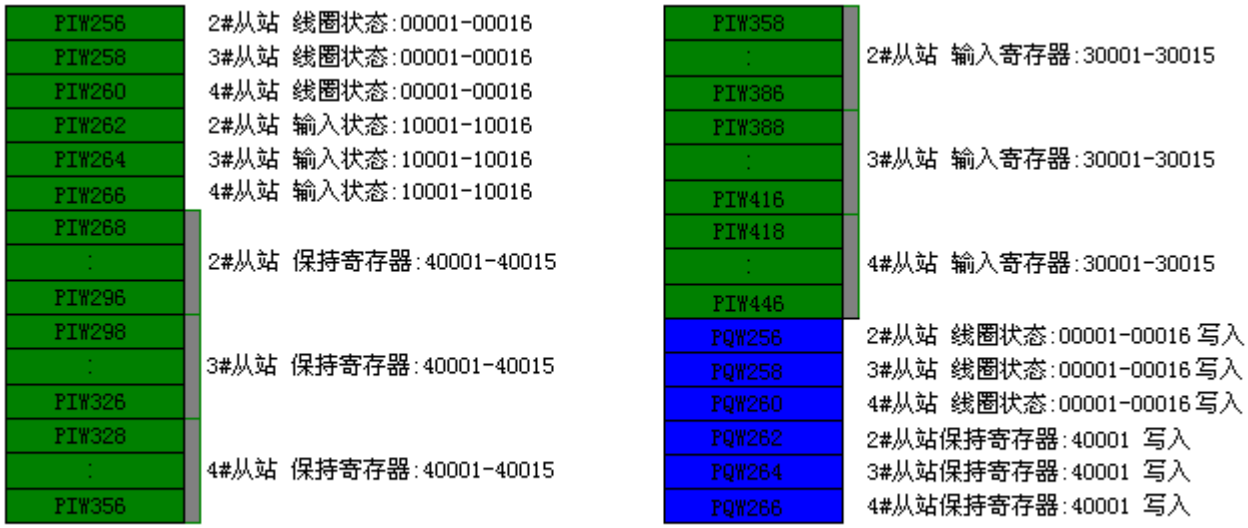


图5 ModbusRTU从站地址映射表

配置 COMMAND 命令表

在 SYCON.net 软件中对 NT30-DPS 进行 Modbus RTU 通讯任务配置时最重要的步骤就是配置 COMMAND 命令表，COMMAND 表最多可以配置 300 个命令。不正确的 COMMAND 表将导致网关无法正常工作，Profibus-DP 总线也无法建立连接，因此一定要搞清楚 COMMAND 表的参数配置意义！

下面是按照以上通讯任务建立的 COMMAND 表（其中 Slave 列至 Write 列为 COMMAND 表中的列，编号和注释栏是额外增加用于描述的）。

COMMAND 表

编号	从站	功能号	地址	数量	寄存器	位	写	注释
ID	Slave	Function	Address	Quantity	Register	.Coil	Write	Comment
1	2	1	1	1	1	1	Cycle	任务 1: 用 FC1 读取 2#从站的 00001 至 00016 共 16 个输出线圈的状态，并把它们放入输入映像区 1#寄存器的 1-16 位中，其中 Quantity 代表一个位。（PIW256）
2	2	1	2	1	1	2	Cycle	
3	2	1	3	1	1	3	Cycle	
4	2	1	4	1	1	4	Cycle	
5	2	1	5	1	1	5	Cycle	
6	2	1	6	1	1	6	Cycle	
7	2	1	7	1	1	7	Cycle	
8	2	1	8	1	1	8	Cycle	
9	2	1	9	1	1	9	Cycle	
10	2	1	10	1	1	10	Cycle	
11	2	1	11	1	1	11	Cycle	
12	2	1	12	1	1	12	Cycle	
13	2	1	13	1	1	13	Cycle	
14	2	1	14	1	1	14	Cycle	
15	2	1	15	1	1	15	Cycle	
16	2	1	16	1	1	16	Cycle	
17	3	1	1	1	2	1	Cycle	任务 1: 用 FC1 读取 3#从站的 00001 至 00016 共 16 个输出线圈的状态，并把它们放入输入映像区 2#寄存器的 1-16 位中，其中 Quantity 代表一个位。（PIW258）
18	3	1	2	1	2	2	Cycle	
19	3	1	3	1	2	3	Cycle	
20	3	1	4	1	2	4	Cycle	
21	3	1	5	1	2	5	Cycle	
22	3	1	6	1	2	6	Cycle	
23	3	1	7	1	2	7	Cycle	
24	3	1	8	1	2	8	Cycle	
25	3	1	9	1	2	9	Cycle	
26	3	1	10	1	2	10	Cycle	
27	3	1	11	1	2	11	Cycle	
28	3	1	12	1	2	12	Cycle	
29	3	1	13	1	2	13	Cycle	
30	3	1	14	1	2	14	Cycle	
31	3	1	15	1	2	15	Cycle	
32	3	1	16	1	2	16	Cycle	
33	4	1	1	1	3	1	Cycle	任务 1: 用 FC1 读取 4#从站的 00001 至 00016 共 16 个输出线圈的状态，并把它们放入输入映像区 3#寄存器的 1-16 位中，其中 Quantity 代表一个位。（PIW260）
34	4	1	2	1	3	2	Cycle	
35	4	1	3	1	3	3	Cycle	
36	4	1	4	1	3	4	Cycle	
37	4	1	5	1	3	5	Cycle	
38	4	1	6	1	3	6	Cycle	
39	4	1	7	1	3	7	Cycle	
40	4	1	8	1	3	8	Cycle	
41	4	1	9	1	3	9	Cycle	
42	4	1	10	1	3	10	Cycle	
43	4	1	11	1	3	11	Cycle	
44	4	1	12	1	3	12	Cycle	
45	4	1	13	1	3	13	Cycle	
46	4	1	14	1	3	14	Cycle	
47	4	1	15	1	3	15	Cycle	
48	4	1	16	1	3	16	Cycle	
49	2	2	10001	1	4	1	Cycle	任务 2: 用 FC2 读取 2#从站的 10001 至 10016 共 16 个输入状态，并把它们放入输入映像区 4#寄存器的 1-16 位中，其中 Quantity 代表一个位。（PIW262）
50	2	2	10002	1	4	2	Cycle	
51	2	2	10003	1	4	3	Cycle	
52	2	2	10004	1	4	4	Cycle	
53	2	2	10005	1	4	5	Cycle	
54	2	2	10006	1	4	6	Cycle	
55	2	2	10007	1	4	7	Cycle	
56	2	2	10008	1	4	8	Cycle	
57	2	2	10009	1	4	9	Cycle	
58	2	2	10010	1	4	10	Cycle	

59	2	2	10011	1	4	11	Cycle	
60	2	2	10012	1	4	12	Cycle	
61	2	2	10013	1	4	13	Cycle	
62	2	2	10014	1	4	14	Cycle	
63	2	2	10015	1	4	15	Cycle	
64	2	2	10016	1	4	16	Cycle	
65	3	2	10001	1	5	1	Cycle	任务 2： 用 FC2 读取 3#从站的 10001 至 10016 共 16 个输入状态，并把它们放入输入映像区 5#寄存器的 1-16 位中，其中 Quantity 代表一个位。（PIW264）
66	3	2	10002	1	5	2	Cycle	
67	3	2	10003	1	5	3	Cycle	
68	3	2	10004	1	5	4	Cycle	
69	3	2	10005	1	5	5	Cycle	
70	3	2	10006	1	5	6	Cycle	
71	3	2	10007	1	5	7	Cycle	
72	3	2	10008	1	5	8	Cycle	
73	3	2	10009	1	5	9	Cycle	
74	3	2	10010	1	5	10	Cycle	
75	3	2	10011	1	5	11	Cycle	
76	3	2	10012	1	5	12	Cycle	
77	3	2	10013	1	5	13	Cycle	
78	3	2	10014	1	5	14	Cycle	
79	3	2	10015	1	5	15	Cycle	
80	3	2	10016	1	5	16	Cycle	
81	4	2	10001	1	6	1	Cycle	任务 2： 用 FC2 读取 4#从站的 10001 至 10016 共 16 个输入状态，并把它们放入输入映像区 6#寄存器的 1-16 位中，其中 Quantity 代表一个位。（PIW266）
82	4	2	10002	1	6	2	Cycle	
83	4	2	10003	1	6	3	Cycle	
84	4	2	10004	1	6	4	Cycle	
85	4	2	10005	1	6	5	Cycle	
86	4	2	10006	1	6	6	Cycle	
87	4	2	10007	1	6	7	Cycle	
88	4	2	10008	1	6	8	Cycle	
89	4	2	10009	1	6	9	Cycle	
90	4	2	10010	1	6	10	Cycle	
91	4	2	10011	1	6	11	Cycle	
92	4	2	10012	1	6	12	Cycle	
93	4	2	10013	1	6	13	Cycle	
94	4	2	10014	1	6	14	Cycle	
95	4	2	10015	1	6	15	Cycle	
96	4	2	10016	1	6	16	Cycle	
97	2	3	40001	15	7	0	Cycle	任务 3： 用 FC3 分三次读取 2#-4#从站的 40001-40015 共 15 个保持寄存器并把它们放入输入映像区 7#寄存器开始的区域（PIW268-PIW356） 注意：寄存器区域不能重叠并且 Coil=0
98	3	3	40001	15	22	0	Cycle	
99	4	3	40001	15	37	0	Cycle	
100	2	4	30001	15	52	0	Cycle	任务 4： 用 FC4 分三次读取 2#-4#从站的 30001-30015 共 15 个输入寄存器并把它们放入输入映像区 52#寄存器开始的区域（PIW358-PIW446） 注意：寄存器区域不能重叠并且 Coil=0
101	3	4	30001	15	67	0	Cycle	
102	4	4	30001	15	82	0	Cycle	
103	2	5	1	1	1	1	Cycle	任务 5： 用 FC5 将输出映像区 1#寄存器的 16 个位值写入 2#从站的 00001 至 00016 共 16 个线圈状态（PQW256）
104	2	5	2	1	1	2	Cycle	
105	2	5	3	1	1	3	Cycle	
106	2	5	4	1	1	4	Cycle	
107	2	5	5	1	1	5	Cycle	

108	2	5	6	1	1	6	Cycle	
109	2	5	7	1	1	7	Cycle	
110	2	5	8	1	1	8	Cycle	
111	2	5	9	1	1	9	Cycle	
112	2	5	10	1	1	10	Cycle	
113	2	5	11	1	1	11	Cycle	
114	2	5	12	1	1	12	Cycle	
115	2	5	13	1	1	13	Cycle	
116	2	5	14	1	1	14	Cycle	
117	2	5	15	1	1	15	Cycle	
118	2	5	16	1	1	16	Cycle	
119	3	5	1	1	2	1	Cycle	任务 5： 用 FC5 将输出映像区 2#寄存器的 16 个位值写入 3#从站的 00001 至 00016 共 16 个线圈状态，其中 Quantity 代表一个位。（PQW258）
120	3	5	2	1	2	2	Cycle	
121	3	5	3	1	2	3	Cycle	
122	3	5	4	1	2	4	Cycle	
123	3	5	5	1	2	5	Cycle	
124	3	5	6	1	2	6	Cycle	
125	3	5	7	1	2	7	Cycle	
126	3	5	8	1	2	8	Cycle	
127	3	5	9	1	2	9	Cycle	
128	3	5	10	1	2	10	Cycle	
129	3	5	11	1	2	11	Cycle	
130	3	5	12	1	2	12	Cycle	
131	3	5	13	1	2	13	Cycle	
132	3	5	14	1	2	14	Cycle	
133	3	5	15	1	2	15	Cycle	
134	3	5	16	1	2	16	Cycle	
135	4	5	1	1	3	1	Cycle	任务 5： 用 FC5 将输出映像区 3#寄存器的 16 个位值写入 4#从站的 00001 至 00016 共 16 个线圈状态，其中 Quantity 代表一个位。（PQW260）
136	4	5	2	1	3	2	Cycle	
137	4	5	3	1	3	3	Cycle	
138	4	5	4	1	3	4	Cycle	
139	4	5	5	1	3	5	Cycle	
140	4	5	6	1	3	6	Cycle	
141	4	5	7	1	3	7	Cycle	
142	4	5	8	1	3	8	Cycle	
143	4	5	9	1	3	9	Cycle	
144	4	5	10	1	3	10	Cycle	
145	4	5	11	1	3	11	Cycle	
146	4	5	12	1	3	12	Cycle	
147	4	5	13	1	3	13	Cycle	
148	4	5	14	1	3	14	Cycle	
149	4	5	15	1	3	15	Cycle	
150	4	5	16	1	3	16	Cycle	
151	2	6	40001	1	4	0	Cycle	任务 6： 用 FC6 分三次将输出映像区 4#寄存器开始的值依次写入 2#-4#从站的 40001 保持寄存器中（PQW262-PQW266） 注意：寄存器区域不能重叠并且 Coil=0
152	3	6	40001	1	5	0	Cycle	
153	4	6	40001	1	6	0	Cycle	

配置 COMMAND 表的注意点：

① 用读取命令 FC1,FC2,FC3,FC4 读取的数据存放到输入映像区，并且从 1 号寄存器开始；

② 用写入命令 FC5,FC6 写入的数据源来自输出映像区，并且从 1 号寄存器开始；

③ 用位读写命令 FC1,FC2,FC5 时要注意将 Modbus 位地址与某寄存器的位号（Coil，1-16）相对应，一个寄存器可以对应 16 个位；

④ 用字读写命令 FC3,FC4,FC6 时要注意将寄存器的位号 Coil 设置为 0；

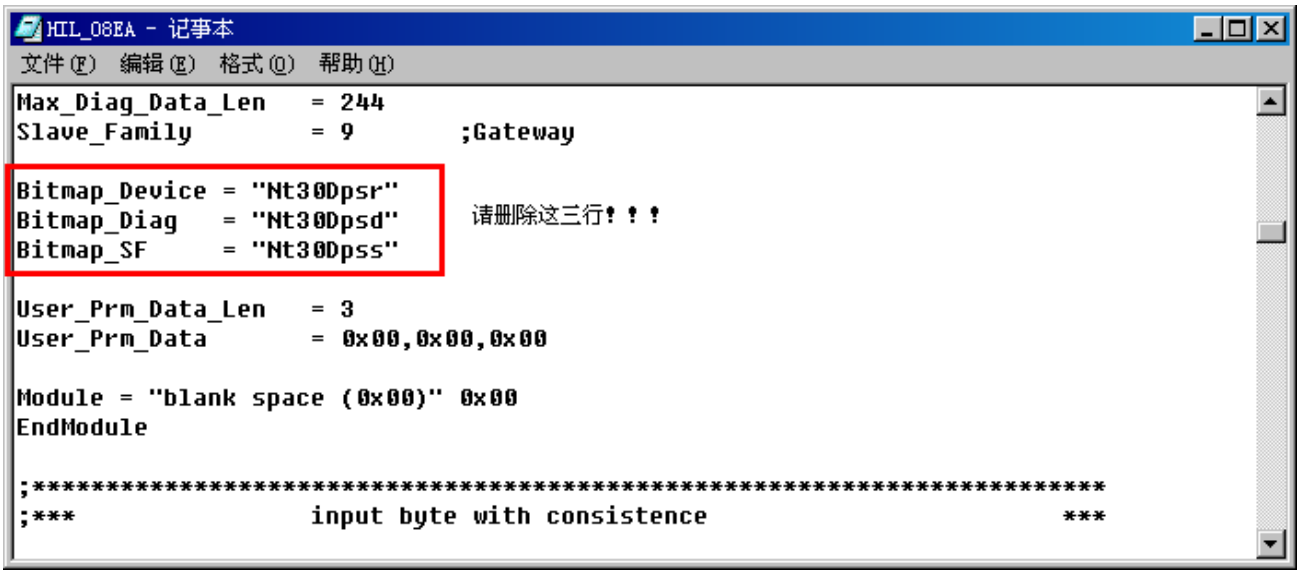
⑤ 输入输出映像区的寄存器编号不能重叠，否则网关自检配置不通过，网关不会进入 Profibus-DP Active 状态；

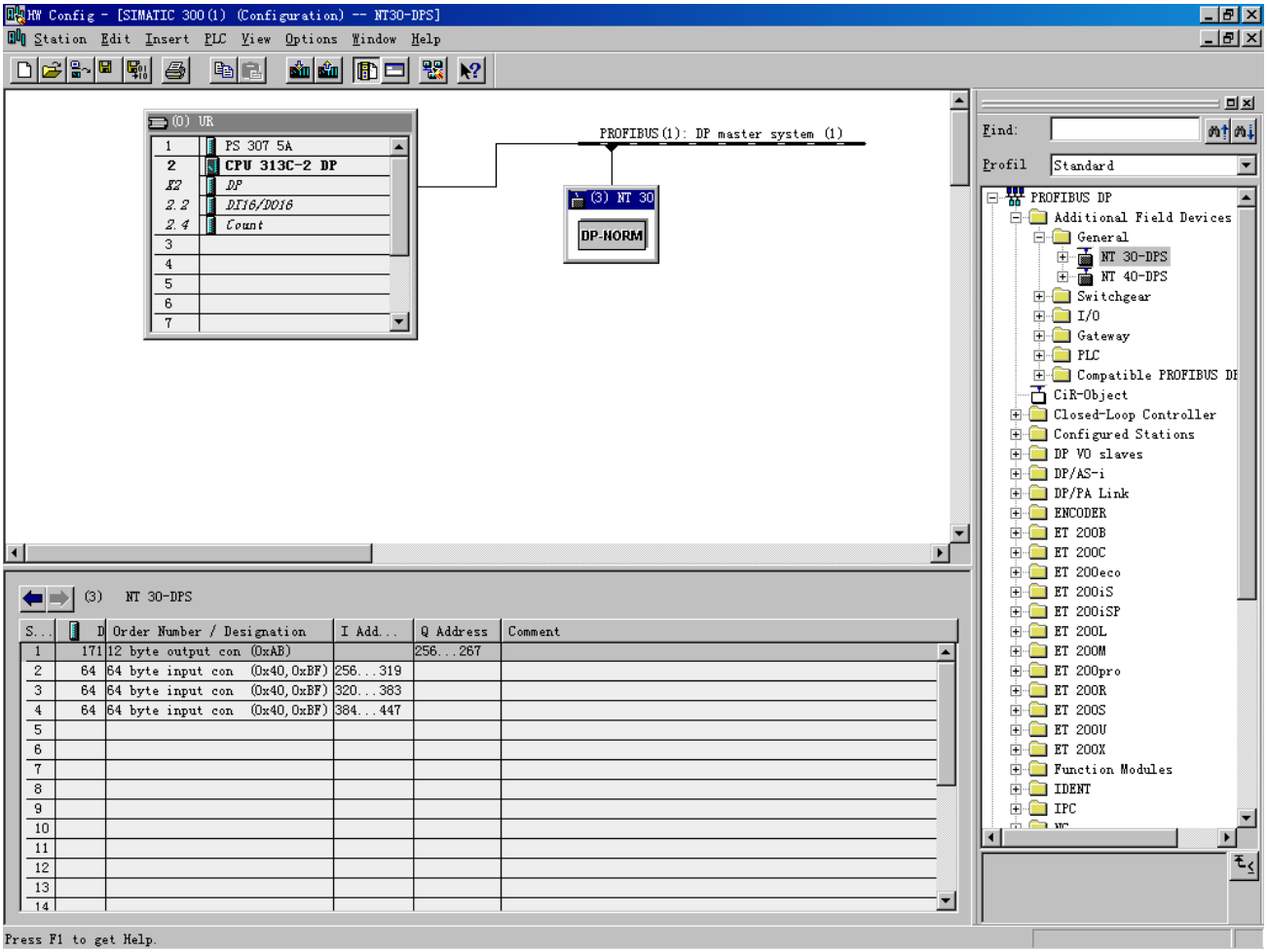
理解了 Profibus-DP 映像区和 COMMAND 表后就可以开始具体的应用配置了，下面是演示的具体步骤。

应用步骤：

1、 运行 STEP7 软件，新建 S7300 项目，导入 NT30-DPS 的 GSD 文件（HIL_08EA.GSD 和 Hil_0916.GSD，在光盘中的 GSD\PROFIBUS 文件夹下），将 NT30-DPS 挂到 CPU313C-2DP 的 Profibus-DP 总线上，分配 NT30-DPS 的输入输出映像区大小，完成后将硬件配置下载到 PLC。

注意：在导入 HIL_08EA.GSD 和 Hil_0916.GSD 前请用记事本打开这两个文件，并把开头为 Bitmap_*的三行删除后保存，否则导入会出现错误。如下图所示：





按照上面的规划，我们为 NT30-DPS 分配了一个 12 字节的输出映像区 PQW256-PQW266，三个 64 字节的输入映像区 PIW256-PIW318、PIW320-PIW382、PIW384-PIW446，共 192 个字节（在 SYCON.net 中单个地址区域最大为 64 字节）。NT30-DPS 的 Profibus 地址为 3，CPU313C-2DP 的 Profibus 地址为 2。如上图所示。

保存 CPU 的硬件配置并下载到 CPU 中。

由于 CPU 的 PQW 区不能在监控表中直接监控，因此需要在 OB1 中通过其他区域地址进行赋值。在 OB1 中我们通过 M 区进行数据赋值，指令如下：

```
L MW0
T PQW256      //传送 MW0 的值到 PQW256，对应 2#从站的 00001-00016 位值
L MW2
T PQW258      //传送 MW2 的值到 PQW258，对应 3#从站的 00001-00016 位值
L MW4
T PQW260      //传送 MW4 的值到 PQW260，对应 4#从站的 00001-00016 位值
L MW6
T PQW262      //传送 MW6 的值到 PQW262，对应 2#从站的 40001 寄存器值
L MW8
T PQW264      //传送 MW8 的值到 PQW264，对应 3#从站的 40001 寄存器值
L MW10
T PQW266      //传送 MW10 的值到 PQW266，对应 4#从站的 40001 寄存器值
```

保存 OB1 并下载到 CPU 中。

2、 将配置电缆 NT-DIAG-RS（配置电缆在网关购买时与网关一起提供）一端接到计算机的串口 COM2，在网关断电状态下将另一端插入网关的配置插槽。运行 SYCON.net 软件，从软件窗口右边产品列表中选择 NT30-DPS-RS2 并拖拉到左边，双击网关图标进入配置窗口，首先通过 Device Assignment 查找网关，找到后选择网关；然后点击 Firmware，选择固件 NTDPSMBR，并点击 Download 下载到网关（应用网关前必须先下载固件！）。接下来就可以配置参数了，修改以下参数，其他默认：

① MODBUS 参数表

Parity=none //无校验，这个参数要和 ModSim32 仿真软件中的设置一致，在实际应用时要和站点配置一致

Mode=master / Address:40001-49999 //连接的是 ModbusRTU 从站，所以网关是主站，地址遵循标准格式

Modbus address=1 //网关的 Modbus 地址=1

② MODULES 参数表

out byte con=12 //12 字节的输出映像区，必须和 STEP7 硬件配置一致

in byte con=64 //64 字节的输入映像区，必须和 STEP7 硬件配置一致

in byte con=64 //64 字节的输入映像区，必须和 STEP7 硬件配置一致

in byte con=64 //64 字节的输入映像区，必须和 STEP7 硬件配置一致

③ COMMAND 参数表：按照上面的 COMMAND 表内容输入；

④ SUPERVIS 参数表（从站状态监视表）：

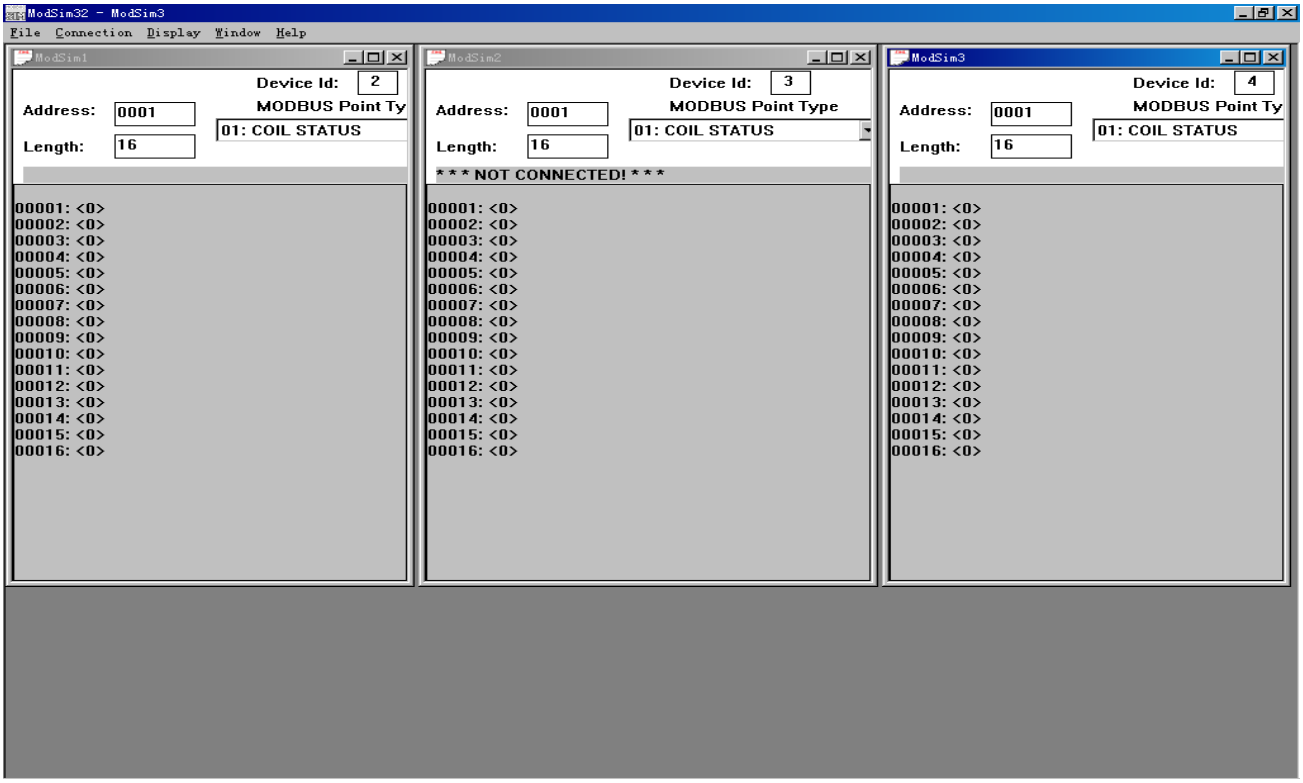
Supervision Mode=SlaveError //启动从站错误监视

注意：这里必须选择任意一个错误模式，但不能为 off。如果为 off，那么网关在遇到读写错误时将不会继续执行下一个命令！SUPERVIS 中的其他参数不需要设定，默认即可；

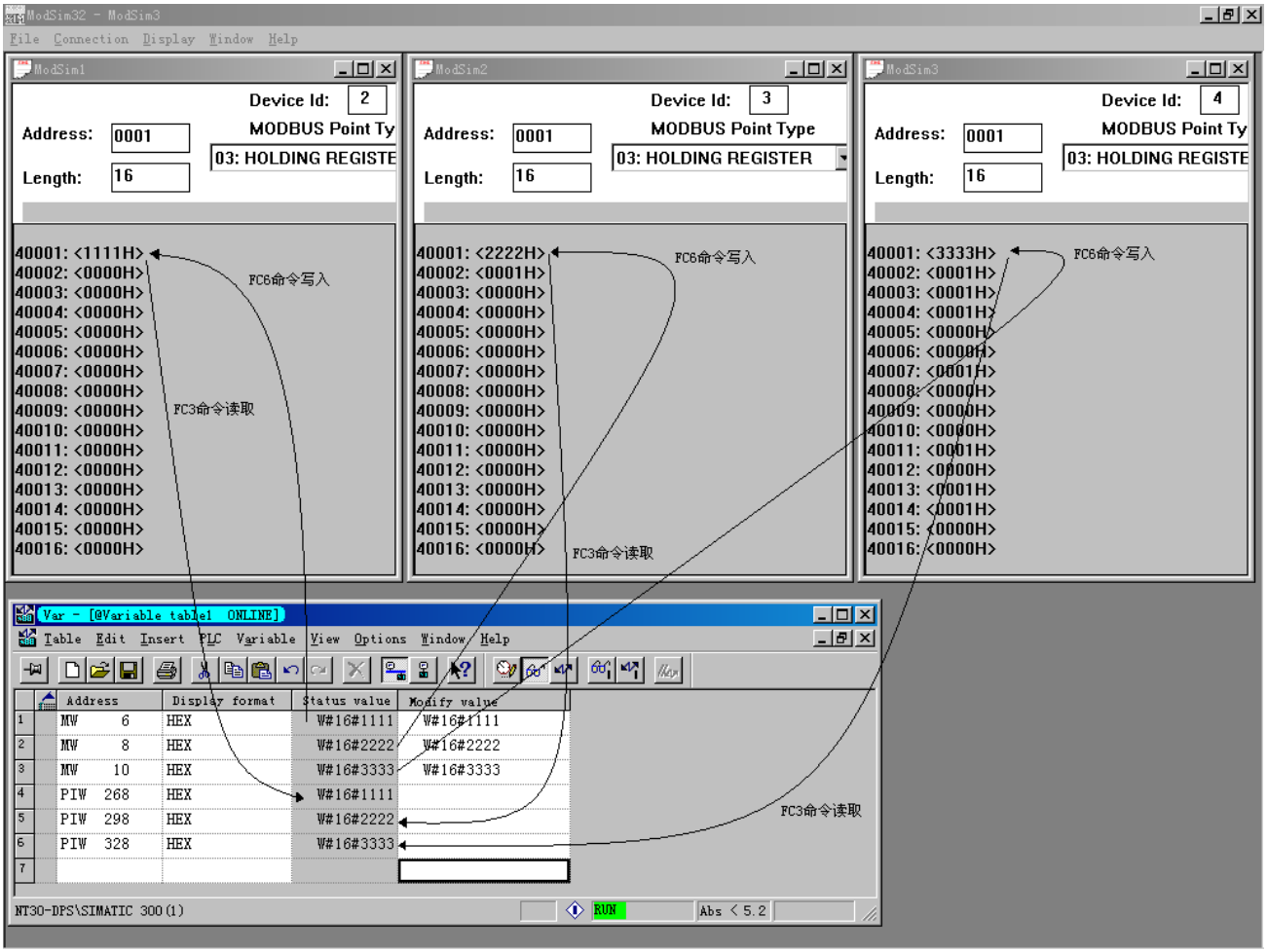
关闭配置窗口并且保存，然后将配置下载到网关中；在网关面板上设定 Profibus-DP 从站地址 3，然后重新将网关上电。如果配置正确，那么可以看到网关的 STA 指示灯常亮，这说明网关已经和 CPU 建立了 Profibus-DP 连接，网关已经开始运行，关闭 SYCON.net 软件。

3、 从计算机的串口 COM2 端口移除网关的诊断电缆，将网关的串口引出线通过 RS232 转接电缆接入 COM2，运行

ModSim32 仿真软件，新建三个从站，Device ID 分别为 2,3,4，如下图所示：



下图为仿真界面



新建从站 2,3,4 后选择菜单 Connection->Connect->Port2，连接到 Port2，即 COM2 口。注意设定 Parity=NONE。建立连接后在 ModSim32 仿真软件中修改每个站点的各区域地址的数值，然后在 STEP7 的监控表中就可以观察到 PIW 各对应区域的数值了。对于任务 3 和任务 6，我们也可以看到 PIW 区域的数值与 PQW 区域的数值是一致的（由于命令较多，数值的读取可能有些延时）：

① 通过 MW6 写入到 PQW262 的数值 16#1111（任务 6，对应 2#从站的 40001 寄存器数据）被 PIW268（任务 3）读到；

② 通过 MW8 写入到 PQW264 的数值 16#2222（任务 6，对应 3#从站的 40001 寄存器数据）被 PIW298（任务 3）读到；

③ 通过 MW10 写入到 PQW266 的数值 16#3333（任务 6，对应 4#从站的 40001 寄存器数据）被 PIW328（任务 3）读到；

如上图仿真界面所示。

六、总结

NT30-DPS 网关将 ModbusRTU 设备接入 Profibus-DP 网络，NT30-DPS 既可以设置成 ModbusRTU 主站用来读写外部 ModbusRTU 从站，也可以设置成 ModbusRTU 从站用来接受外部 ModbusRTU 主站的读写访问。在 NT30-DPS 内部的 Modbus 地址数据被映射到网关的 Profibus-DP 的输入输出映像区，因此对于 Profibus-DP 的主站 CPU 来说可以很方便的直接读写 Modbus 地址数据而不需要考虑任何 ModbusRTU 协议细节，从而在工程应用上实现快速的 ModbusRTU 设备的联网控制。

欢迎您随时咨询我公司，对于您的联网要求，我们有能力为您提供最佳的工业联网控制解决方案。

无锡市北辰自动化技术有限公司
地址：江苏省无锡市滴翠路 100 号国家工业设计园创意园 A 幢 903
电话：0510-85166823,85162669
传真：0510-85166813
邮箱：beichen@beichen-automation.com
网址：www.beichen-automation.com
中华工控网网址：www.gkong.com/co/beichen-automation/